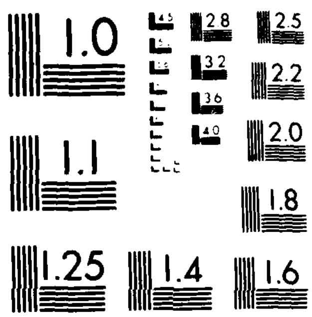
AD-A128 491 GRID GENERATION ABOUT A FIN-CYLINDER COMBINATION(U) 1/1
PENNSYLVANIA STATE UNIV UNIVERSITY PARK APPLIED
RESEARCH LAB G H HOFFMAN 30 MAR 83 ARL/PSU/TM-83-45
UNCLASSIFIED N00024-79-C-6043 F/G 12/1 NL

END
DATE
FILMED
6-83
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0

(6)

GRID GENERATION ABOUT A FIN-CYLINDER
COMBINATION

G. H. Hoffman

Technical Memorandum
File No. TM-83-45
30 March 1983
Contract No. N00024-79-C-6043

Copy No. 7

The Pennsylvania State University
Intercollege Research Programs and Facilities
APPLIED RESEARCH LABORATORY
Post Office Box 30
State College, PA 16801

Approved for Public Release
Distribution Unlimited

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: TM 83-45	2. GOVT ACCESSION NO.: AD-A128491
3. RECIPIENT'S CATALOG NUMBER:	
4. TITLE (and Subtitle): GRID GENERATION ABOUT A FIN-CYLINDER COMBINATION	5. TYPE OF REPORT & PERIOD COVERED: Technical Memorandum
	6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): G. H. Hoffman	8. CONTRACT OR GRANT NUMBER(s): N00024-79-C-6043
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Applied Research Laboratory, Post Office Box 30, State College, PA 16801	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Sea Systems Command, Code NSEA-63R31, Washington, DC 20362	12. REPORT DATE: 30 March 1983
	13. NUMBER OF PAGES: 56
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report)
	15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release. Distribution unlimited.
Per NAVSEA -

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

three-dimensional, grid, generation, cylinder

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An algebraic grid generation procedure is presented which produces a three-dimensional, body fitted coordinate system about a right circular cylinder with four symmetric fins attached. Special features of the grid are an initial value plane normal to the cylinder axis and the ability to cluster lines near the fin and cylinder surfaces for viscous/turbulent flow calculations. The method used is a modification of the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Jameson-Caughey procedure developed originally for inviscid transonic flow calculations about wing-fuselage combinations. In this procedure, a sequence of conformal transformations followed by a shearing transformation is used to map the irregular flow domain in physical space into a rectangular shaped computational domain. A three-dimensional grid is produced by stacking two-dimensional mappings. The method is therefore extremely fast. The main features of the procedure are discussed and two numerical examples of grids are presented for a fin composed of a symmetric Joukowsky airfoil.

Subject: Grid Generation about a Fin-Cylinder Combination

References: See Page 29

Abstract: An algebraic grid generation procedure is presented which produces a three-dimensional, body fitted coordinate system about a right circular cylinder with four symmetric fins attached. Special features of the grid are an initial value plane normal to the cylinder axis and the ability to cluster lines near the fin and cylinder surfaces for viscous/turbulent flow calculations. The method used is a modification of the Jameson-Caughey procedure developed originally for inviscid transonic flow calculations about wing-fuselage combinations. In this procedure, a sequence of conformal transformations followed by a shearing transformation is used to map the irregular flow domain in physical space into a rectangular shaped computational domain. A three-dimensional grid is produced by stacking two-dimensional mappings. The method is therefore extremely fast. The main features of the procedure are discussed and two numerical examples of grids are presented for a fin composed of a symmetric Joukowsky airfoil.

Acknowledgment: This work was sponsored by the Office of Naval Research under the Fundamental Research Initiatives Program.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. INTRODUCTION

This report addresses the problem of generating a surface-fitted grid in a model fin-body problem consisting of a circular cylinder with four identical symmetric fins attached. This grid is to be used in the calculation of incompressible, laminar flow at moderate-to-high Reynolds numbers. The aim of the calculation is to resolve the details of the separated zone at the leading edge of the fin-cylinder juncture and the subsequent vortex that forms downstream. Thus, the grid must have proper clustering so as to resolve the regions of high flow gradients.

The approach used here is to generate the grid analytically but to determine the metric coefficients numerically. Such an approach has been pursued successfully by Jameson [1] and Caughey and Jameson [2-4] in solving three-dimensional inviscid transonic flows about wing-body combinations. The basic idea is to map the physical geometry to a strip of almost constant width using a sequence of conformal transformations. Then boundary fitted coordinates are generated by the application of a shearing transformation. The result of the latter transformation is a nonorthogonal coordinate system in the physical plane but one in which the nonorthogonality can be controlled.

The present work is an extension of the Jameson-Caughey technique for what is called the wind tunnel problem to the case of an initial value plane ahead of the airfoil. In order to treat viscous flow, clustering transformations are used so that the computational grid is uniform in all three directions.

One advantage of the present technique is that, owing to the simple cylindrical body geometry, a three-dimensional grid is generated by stacking a series of two-dimensional grids. Another advantage of the analytical approach over the numerical solution of elliptic partial differential equations as a means of grid generation is its much greater speed which is especially important for three-dimensional applications.

II. ANALYSIS

2.1 Geometry of Computational Domain

We start the grid generation analysis by defining the geometry about which a surface fitted grid is to be generated and the extent of the computational domain.

1. The body is an infinitely long, hollow circular cylinder of radius R_c with its centerline parallel to the free-stream velocity vector.
2. Four identical fins of constant unit chord and infinite span, consisting of symmetric airfoil sections, are mounted on the cylinder 90 degrees apart with their chord planes passing through the cylinder axis.
3. The computational domain consists of the region interior to an outer cylinder of radius R_t which encases the inner cylinder and fins, bounded upstream and downstream by planes normal to the cylinder axis.

A schematic of one fourth of the geometry and computational domain is shown in Fig. 1 and a head-on view showing the coordinate system in the crossflow plane appears in Fig. 2. Since the fins are identical and equally spaced, we have four planes of symmetry, namely, at $\theta = 0$, $\pi/4$, $\pi/2$ and $3\pi/4$. Thus, in the flow field calculation for this model problem only the segment $0 \leq \theta \leq \pi/4$ needs to be considered.

2.2 Sequence of Transformations

Four transformations applied in sequence are required to map the fin-cylinder and surrounding computational domain into a rectangular parallelpiped. Then a fifth stretching transformation is applied to adjust the grid line spacings for proper flow field resolution in physical space and to allow a uniform step size in all three computational coordinates.

We start by defining polar coordinates (r,θ) in the crossflow plane, as shown in Fig. 2, according to

$$r = (y^2 + z^2)^{1/2} \quad , \tag{1}$$

$$\theta = \tan^{-1}\left(\frac{y}{z}\right) \quad . \tag{2}$$

Thus, points in physical space are defined by standard cylindrical coordinates (x,r,θ).

Following Caughey and Jameson [2], the first transformation normalizes (x,r,θ) according to (all lengths are referred to the airfoil chord):

$$\bar{x} = x - d_s + \ln 2 \quad , \tag{3}$$

$$\bar{r} = \frac{r - R_c}{R_t - R_c} \quad , \tag{4}$$

$$\bar{\theta} = 4\theta \quad , \tag{5}$$

where d_s is the location of the singular point of the unwrapping transformation and is just inside the leading edge of the airfoil. Note that in the above definitions, $0 < \bar{r} < 1$ and $0 < \bar{\theta} < \pi$ in the computational domain. The upper limit on $\bar{\theta}$ is convenient in the next transformation.

Because $\bar{r}$ = constant is a surface fitted coordinate we need only generate a surface fitted grid in the $(\bar{x},\bar{\theta})$ plane. The geometry of an $\bar{r}$ = constant surface in the computational domain is sketched in Fig. 3.

The conformal transformation

$$\bar{x} - i\bar{\theta} = \ln[1 - \cosh(\xi + i\eta)] \;, \tag{6}$$

applied to an $\bar{r}$ = constant surface unwraps the geometry in Fig. 3 to produce the domain shown in Fig. 4. The minus sign has been used on the left in Eq. (6) so that the upper symmetry plane maps to the positive ξ axis.

In the present problem initial conditions from an axis etric boundary layer-potential flow composite solution are specified on tl lane $\bar{x} = -\bar{a}$. This initial value line in an $\bar{r}$ = constant surface (IVL) is own as line segment ABC in Fig. 3. Under transformation (6), the IVL m o a near semi-circle in the (ξ,η) plane, as shown in Fig. 4. The airfoil image in this plane is the arc DEF.

We next apply another conformal transformation to nearly straighten out the IVL in Fig. 4. This transformation is

$$\bar{\xi} + i\bar{\eta} = \xi + i\eta + \frac{\xi_o^2}{\xi + i\eta} \;, \tag{7}$$

where ξ_o is the intersection of the IVL with the ξ axis (Point A in Fig. 4). The conformal transformation (7) maps the upper and lower boundaries in the (ξ,η) plane into slowly varying functions of $\bar{\xi}$ in the $(\bar{\xi},\bar{\eta})$ plane, as shown in Fig. 5. We note that near Points A and C the IVL is now cusp-like.

The fourth transformation is a shearing transformation which straightens out the upper and lower boundaries in the $(\bar{\xi},\bar{\eta})$ plane. This transformation is

$$X = \bar{\xi} \ , \tag{8}$$

$$Y = \frac{\bar{\eta} - \bar{\eta}_L}{D} \ , \tag{9}$$

$$Z = \bar{r} \ , \tag{10}$$

where

$$D \equiv D(\bar{\xi},\bar{r}) = \bar{\eta}_U - \bar{\eta}_L \ , \tag{11}$$

and $\bar{\eta}_U$ and $\bar{\eta}_L$ are the ordinates, at a given $\bar{\xi}$, of the upper and lower boundaries in the $(\bar{\xi},\bar{\eta})$ plane.

Finally, to provide for clustering the grid lines near the fin and cylinder surfaces to resolve the viscous layers there and to space lines around the airfoil and in the wake as desired, we introduce one-dimensional stretching functions as follows:

$$X_c = F_1(X) \ , \tag{12}$$

$$Y_c = F_2(Y) \ , \tag{13}$$

$$Z_c = F_3(Z) \ , \tag{14}$$

For the time being we leave F_1, F_2 and F_3 unspecified. Thus (X_c, Y_c, Z_c) are the computational coordinates devised so that the step sizes ΔX_c, ΔY_c and ΔZ_c are constants.

2.3 Conformal Mapping Relations

Since the FORTRAN code is written in terms of real variables, the real and imaginary parts of the conformal mappings must be determined. In addition, the inverses of both mappings are needed because the grid generation procedure requires being able to proceed from the $(\bar{x},\bar{\theta})$ plane to the (X_c, Y_c) plane and then back to the $(\bar{x},\bar{\theta})$ plane.

The real and imaginary parts of Eq. (6) yield the two relations:

$$\cosh \xi \cos \eta = 1 - e^{\bar{x}} \cos \bar{\theta} , \tag{15}$$

$$\sinh \xi \sin \eta = e^{\bar{x}} \sin \bar{\theta} . \tag{16}$$

The solutions for $\bar{x}$ and $\bar{\theta}$ are obtained by squaring (15) and (16), then adding and making use of the ordinary and hyperbolic trigonometric identities. The result for $\bar{x}$, choosing the proper sign, is

$$\bar{x} = \ell n(\cosh \xi - \cos \eta) , \tag{17}$$

and $\bar{\theta}$ is obtained from Eq. (15), viz.,

$$\bar{\theta} = \cos^{-1} \left\{ \frac{1 - \cosh \xi \cos \eta}{\cosh \xi - \cos \eta} \right\} . \tag{18}$$

To obtain the solutions for ξ and η we first define,

$$\bar{p} = 1 - e^{\bar{x}} \cos \bar{\theta} , \tag{19}$$

$$\bar{q} = e^{\bar{x}} \sin \bar{\theta} . \tag{20}$$

Following the same procedures as above, we eliminate η to obtain a quadratic equation for $\sinh^2 \xi$ which has the solution

$$\sinh^2 \xi = \frac{1}{2} [(\beta^2 + 4\bar{q}^2)^{1/2} - \beta] \tag{21}$$

where

$$\beta = 1 - \bar{p}^2 - \bar{q}^2 . \tag{22}$$

In the right half plane ξ is the positive root of Eq. (21). The expression for η with the proper behavior $(0 < \eta < \pi)$ is obtained from Eq. (15), viz.

$$\eta = \cos^{-1} \left(\frac{\bar{p}}{\cosh \xi} \right) . \tag{23}$$

Next, the real and imaginary parts of Eq. (7) yield

$$\bar{\xi} = \xi \left(1 + \frac{\xi_o{}^2}{\xi^2 + \eta^2} \right) , \tag{24}$$

$$\bar{\eta} = \eta \left(1 - \frac{\xi_o{}^2}{\xi^2 + \eta^2} \right) . \tag{25}$$

We determine ξ_o from Eq. (17) by setting $\bar{x} = - \bar{a} = - a + \ln 2$ and $\eta = 0$. The The result is

$$\xi_o = \cosh^{-1}(1 + 2e^{-a}) . \tag{26}$$

where $a = d_s + d_{IVL}$.

To solve for ξ and η in terms $\bar{\xi}$ and $\bar{\eta}$ we return to the complex form which is written as,

$$w = z + \frac{\xi_o{}^2}{z} , \tag{27}$$

where

$$w = \bar{\xi} + i\bar{\eta} \ , \tag{28}$$

$$z = \xi + i\eta \ . \tag{29}$$

Solving Eq. (27) for z yields,

$$\phi^2 = \frac{1}{4} w^2 - \xi_o{}^2 \ , \tag{30}$$

where

$$\phi = z - \frac{1}{2} w \ . \tag{31}$$

Let us now define

$$\phi = u + iv \ . \tag{32}$$

Then, combining Eqs. (28), (29) and (31) gives:

$$\xi = u + \frac{1}{2} \bar{\xi} \ , \tag{33}$$

$$\eta = v + \frac{1}{2} \bar{\eta} \ . \tag{34}$$

Now Eq. (30) leads to the following relations:

$$u^2 - v^2 = \hat{p} \ , \tag{35}$$

$$uv = \hat{q} \ , \tag{36}$$

where

$$\hat{p} = \frac{1}{4} (\bar{\xi}^2 - \bar{\eta}^2) - \xi_o{}^2 \ , \tag{37}$$

$$\hat{q} = \frac{1}{4} \bar{\xi}\bar{\eta} \ . \tag{38}$$

Equations (35) and (36) can be solved for u and v with the result:

$$u^2 = \frac{1}{2}(\hat{\mu} + \hat{p}) \quad , \tag{39}$$

$$v^2 = \frac{1}{2}(\hat{\mu} - \hat{p}) \quad , \tag{40}$$

where

$$\hat{\mu} = (\hat{p}^2 + 4\hat{q}^2)^{1/2} \; . \tag{41}$$

Then the final result for ξ and η, combining Eqs. (33), (34), (39) and (40), is

$$\xi = \frac{1}{2}\bar{\xi} + [\frac{1}{2}(\hat{\mu} + \hat{p})]^{1/2} \quad , \tag{42}$$

$$\eta = \frac{1}{2}\bar{\eta} + [\frac{1}{2}(\hat{\mu} - \hat{p})]^{1/2} \; . \tag{43}$$

2.4 Calculation of Shearing Boundaries

The shearing boundaries, which are straightened out by the shearing transformation Eq. (9), are defined as $\bar{\eta}_U(\bar{\xi})$ and $\bar{\eta}_L(\bar{\xi})$. Thus $\bar{\eta}_U$ is the image of the upper airfoil surface and the line $\bar{\theta} = 0$ downstream of the trailing edge while $\bar{\eta}_L$ is the image of the upper half of the initial value line $(\bar{x} = -\bar{a})$ and the line $\bar{\theta} = \pi$ for $\bar{x} > -\bar{a}$.

We start by determining the image of the upper half of the airfoil in the $(\bar{\xi},\bar{\eta})$ plane. The airfoil will be given as a set of points $(x_F, y_F)_i$ where for convenience we take the origin at the leading edge. Then the scaled airfoil coordinates in the $(\bar{x},\bar{\theta})$ plane, for a given r, are:

$$\bar{x}_F = x_F + \ell n\ 2 - d_s \; , \tag{44}$$

$$\bar{\theta}_F = 4 \sin^{-1} \left(\frac{y_F}{r}\right) . \tag{45}$$

Next, the image in the (ξ,η) plane is computed from

$$\xi_F = \sinh^{-1} [\tfrac{1}{2} (\alpha - \beta)]^{1/2} , \tag{46}$$

$$\eta_F = \cos^{-1} \left(\frac{\bar{p}}{\cosh \xi_F}\right) , \tag{47}$$

where

$$\alpha = (\beta^2 + 4\bar{q}^2)^{1/2} , \tag{48}$$

and $\bar{p}$, $\bar{q}$ and β are given by Eqs. (19), (20) and (22). Then the image in the $(\bar{\xi},\bar{\eta})$ plane is

$$\bar{\xi}_F = \xi_F(1 + \mu) , \tag{49}$$

$$\bar{\eta}_U = \eta_F(1 - \mu) , \tag{50}$$

and

$$\mu = \frac{\xi_o{}^2}{\xi_F^2 + \eta_F^2} . \tag{51}$$

The upper boundary beyond the airfoil trailing edge is the image of $\bar{\theta} = 0$ which maps to $\eta = \pi$. To calculate $\bar{\eta}_U$ in this region we first compute a uniform point distribution of $\bar{\xi}$ on the interval $(\bar{\xi}_{TE},\bar{\xi}_{max})$. Then corresponding values of ξ are computed by iteration from

$$\bar{\xi}^{(n+1)} = \frac{\bar{\xi}}{1 + \mu_\pi} , \tag{52}$$

where superscript n denotes the iteration number, and

$$\mu_\pi = \frac{\xi_o{}^2}{\pi^2 + \xi^{(n)}} . \tag{53}$$

We note that Eq. (52) converges quite rapidly. With a value of ξ known, $\bar{\eta}_U$ is computed from

$$\bar{\eta}_U = \pi \cdot (1 - \mu) . \tag{54}$$

In the calculation of the lower $\bar{\eta}$ boundary the shearing transformation requires that the same $\bar{\xi}$ distribution be used as was determined for $\bar{\eta}_U$. The lower boundary is computed in two segments, the first on the interval $(0,\bar{\xi}_o)$, where $\bar{\xi}_o$ is the image of ξ_o, and the second on the remaining interval $(\bar{\xi}_o,\bar{\xi}_{max})$.

On the interval $(0,\bar{\xi}_o)$ we calculate ξ and η by iteration from the rapidly convergent formula:

$$\xi^{(n+1)} = \frac{\bar{\xi}}{1 + \mu} , \tag{55}$$

where in this case

$$\mu = \frac{\xi_o{}^2}{(\xi^2 + \eta^2)^{(n)}} , \tag{56}$$

$$\eta = \cos^{-1}(\cosh \xi^{(n)} - 2e^{-a}) . \tag{57}$$

To start the iteration we set $\mu = 1$ in Eq. (55) which from Eq. (56) is seen to be exact at $\xi = \xi_o$. With ξ and η known, $\bar{\eta}_L$ is calculated from

$$\bar{\eta}_L = \eta \cdot (1 - \mu) \ . \tag{58}$$

On the interval $(\bar{\xi}_o, \bar{\xi}_{max})$ we know from Eq. (58) that the image of $\bar{\theta} = \pi$ is

$$\bar{\eta}_L = 0 \ . \tag{59}$$

Thus knowing the distribution of $\bar{\eta}_U$ and $\bar{\eta}_L$ on $(0, \bar{\xi}_{max})$ we can obtain the distribution of the shearing distance D from Eq. (11).

2.5 Stretching Functions

The approach taken here, as already mentioned, is to use one-dimensional stretching functions, as indicated by Eqs. (12), (13) and (14). In the present application the location and length scales of regions of rapid variation of the solution are known beforehand. In a Z = constant plane of the computational domain, as shown in Fig. 6, clustering of Y = constant lines is needed near Y = 1 and 0 to resolve the boundary layer developing on the airfoil and the region around the corner singularity, $\bar{x} = -\bar{a}$, $\bar{\theta} = \pi$, in the physical plane. Thus, for the variable Y a two-sided stretching function is required. Because of the primary viscous layer on the cylinder clustering is needed near Z = 0 which requires a one-sided stretching function for Z. The stretching function for X depends on criteria related to the flow field and the mapping geometry which will be discussed later.

Vinokur [5] has determined approximate criteria for the development of one- and two-sided stretching functions of one variable which give a uniform truncation error independent of the governing differential equation or difference algorithm. He investigates several analytic functions but finds that only tan z, where z is real or pure imaginary, satisfies all of his criteria.

We start with the stretching function for Y and note that both Y and Y_c are normalized variables as required in Vinokur's functions. In the present case, z is taken to be pure imaginary which leads to

$$Y = \frac{\tanh(Y_c \Delta\phi)}{A \sinh \Delta\phi + (1 - A \cosh \Delta\phi) \tanh(Y_c \Delta\phi)} , \tag{60}$$

where

$$A = (S_0/S_1)^{1/2} , \tag{61}$$

$$B = (S_0 S_1)^{1/2} , \tag{62}$$

and S_0 and S_1 are dimensionless slopes defined as

$$S_0 = \frac{dY_c}{dY} (0) ,$$

$$S_1 = \frac{dY_c}{dY} (1) ,$$

which control the clustering at Y = 0 and Y = 1, and $\Delta\phi$ is the solution of the following transcendental equation:

$$B = \frac{\sinh \Delta\phi}{\Delta\phi} . \tag{63}$$

To avoid solving Eq. (63) by iteration, Vinokur determines the following extremely accurate approximate solutions for small and large B:

<u>For B < 2.7829681</u>

$$\Delta\phi = (6\bar{B})^{1/2}(1 - 0.15\bar{B} + 0.057321429\bar{B}^2$$

$$- 0.024907295\bar{B}^3 + 0.0077424461\bar{B}^4$$

$$- 0.0010794123\bar{B}^5) , \qquad (64)$$

where

$$\bar{B} = B - 1 . \qquad (65)$$

<u>For B > 2.7829681</u>

$$\Delta\phi = V + (1 + 1/V)\ell n(2V) - 0.02041793$$

$$+ 0.24902722W + 1.9496443W^2 - 2.6294547W^3$$

$$+ 8.56795911W^4 , \qquad (66)$$

where

$$V = \ell n\ B , \qquad (67)$$

and

$$W = 1/B - 0.028527431 . \qquad (68)$$

An example of this two-sided stretching function for $S_0 = 100$ and $S_1 = 10$ is shown in Fig. 7. For this case, $\Delta\phi$ computed from Eq. (66) is 5.926.

The one-sided counterpart of Eq. (6) is antisymmetric about the mid-point and, in terms of Z and Z_c, is given by

$$Z = 1 + \frac{\tanh\left[\frac{1}{2}\Delta\phi(Z_c - 1)\right]}{\tanh\frac{\Delta\phi}{2}}, \qquad 0 < Z < 1 \quad , \qquad (69)$$

where $\Delta\phi$ is the solution of

$$S_0 = \frac{\sinh \Delta\phi}{\Delta\phi} , \tag{70}$$

and

$$S_0 = \frac{dZ_c}{dZ} (0) .$$

Two examples of this one-sided stretching function, S_0 = 10 and 100, are shown in Fig. 8.

The stretching function in x is required to have the following properties:

(1) It must have the ability to cluster points near the nose of the airfoil to resolve rapid flow field variations in that region.

(2) Control points, where grid lines are required, are the corner, $X = X_o$, and the airfoil trailing edge, $X = X_{TE}$.

(3) Downstream of the airfoil trailing edge where flow gradients are decreasing the step size should gradually increase.

(4) The stretching function should have continuous first derivatives.

(5) For proper flow field resolution, the number of steps on the intervals $(0,X_o)$ and (X_o,X_{TE}) are to be parameters.

The above requirements dictate the stretching function be made up of three piecewise continuous segments on $(0,X_o)$, on (X_o,X_{TE}) and on (X_{TE},X_{max}).

We start by defining variables normalized on the corner location,

$$\hat{X} = \frac{X}{X_o} , \ \hat{X}_c = \frac{X_c}{X_o} .$$

An appropriate stretching function on the first segment is given by Eq. (61) of Vinokur, viz.

$$\hat{X} = \hat{X}_c \left[1 + \frac{1}{2}(\hat{S}_0 - 1)(1 - \hat{X}_c)(2 - \hat{X}_c)\right] , \quad 0 < \hat{X}_c < 1 , \tag{72}$$

where $\hat{S}_0$ is the slope at the origin and is used to control clustering of points in that region. The uniform step size on Segment 1 is given by

$$\Delta\hat{X}_c = \frac{1.0}{N_1} , \tag{73}$$

where N_1 is the number of intervals on Segment 1. We note that $\Delta\hat{X}_c$, as given by Eq. (73), is also the step size on Segments 2 and 3.

On Segment 2, the scaled trailing edge coordinate is given by

$$(\hat{X}_c)_{TE} = 1 + N_2\, \Delta\hat{X}_c , \tag{74}$$

where N_2 is the number of intervals on Segment 2. We note that $(\hat{X}_c)_{TE} \neq \hat{X}_{TE}$. The constraints to be satisfied by the stretching function of Segment 2 are:

$$\left.\begin{aligned} &\hat{X} = 1, \ \hat{X}' = \hat{X}_1' \text{ on } \hat{X}_c = 1 \\ &\hat{X} = \hat{X}_{TE} \text{ on } \hat{X}c = (\hat{X}_c)_{TE} \end{aligned}\right\} , \tag{75}$$

where

$$\hat{X}_1' = \left.\frac{d\hat{X}}{d\hat{X}_c}\right|_{\hat{X}_c = 1}$$

which from Eq. (72) is

$$\hat{X}_1' = \frac{1}{2}(3 - \hat{S}_0) . \tag{73}$$

With three constraints a parabola is appropriate. The resulting stretching function is

$$\hat{X} = 1 + [\hat{X}_1' + A(\hat{X}_c - 1)](\hat{X}_c - 1) \ , \tag{77}$$

where

$$A = \frac{\hat{X}_{TE} - 1 - \hat{X}_1'[(\hat{X}_c)_{TE} - 1]}{[(\hat{X}_c)_{TE} - 1]^2} \ . \tag{78}$$

On Segment 3 a geometric progression is used to increase the step size in $\hat{X}$. Requiring continuity of $\hat{X}$ at the junction with Segment 2, we have

$$\hat{X}_k = \hat{X}_{TE} + \Delta\hat{X}_1 \left(\frac{1 - \hat{C}^{k-1}}{1 - \hat{C}} \right) \ , \ k > 2 \ , \tag{79}$$

where $\hat{C}$ is the constant step size ratio defined by,

$$\hat{C} = \frac{\Delta\hat{X}_k}{\Delta\hat{X}_{k-1}} > 1 \ .$$

Continuity of the first derivative at the junction is ensured by choosing $\Delta\hat{X}_1$ equal to the last $\Delta\hat{X}$ on Segment 2. No attempt is made to match $\hat{X}_{max}$ exactly.

The stretching function for $\hat{X}$ is seen to have four parameters, $\hat{S}_0$, N_1, N_2 and $\hat{C}$, which provide considerable flexibility in the point distribution of $\hat{X}$. A typical example is shown in Fig. 9.

3. RESULTS AND DISCUSSION

3.1 Generation of the Grid

The step-by-step procedure to generate a grid in the physical plane for a given airfoil shape and initial value plane location is as follows:

(1) The uniform computational grid $(X_{c_i}, Y_{c_j}, Z_{c_k})$ is first established and then (X_i, Y_j, Z_k) are calculated via the stretching functions described in Section 2.5.

(2) With $(X_i\ Y_j, Z_k)$ known, $\bar{r}_k$ is determined from

$$\bar{r}_k = Z_k \ . \tag{80}$$

Then for $\bar{r}$ fixed, the points in the (X,Y) plane are transformed to the $(\bar{\xi},\bar{\eta})$ plane by

$$\bar{\xi}_{ijk} = X_i \ , \tag{81}$$

$$\bar{\eta}_{ijk} = Y_j D_{ik} + (\bar{\eta}_L)_{ik} \ , \tag{82}$$

where

$$D_{ik} = (\bar{\eta}_U)_{ik} - (\bar{\eta}_L)_{ik} \ . \tag{83}$$

By Eq. (45), $\bar{\theta}_F$ depends on r and hence $\bar{r}$ and therefore $\bar{\eta}_L$ and $\bar{\eta}_U$ must be computed anew for each value of $\bar{r}$. The procedure used here is to calculate more points than needed on the shearing boundaries for a given $\bar{r}$ and then to use Lagrange cubic interpolation to determine $\bar{\eta}_L$ and $\bar{\eta}_U$ for a given $\bar{\xi}$.

(3) With $(\bar{\xi},\bar{\eta})$ known, the transformation to the (ξ,η) plane is

$$\xi_{ijk} = \frac{1}{2}\,\bar{\xi}_{ijk} + \left[\frac{1}{2}\,(\hat{\mu} + \hat{p})\right]^{1/2}_{ijk} \;, \tag{84}$$

$$\eta_{ijk} = \frac{1}{2}\,\bar{\eta}_{ijk} + \left[\frac{1}{2}\,(\hat{\mu} - \hat{p})\right]^{1/2}_{ijk} \;, \tag{85}$$

where

$$\hat{\mu} = (\hat{p}^2 + 4\hat{q}^2)^{1/2} \;, \tag{86}$$

$$\hat{p} = \frac{1}{4}\,(\bar{\xi}^2 - \bar{\eta}^2) - \xi_o{}^2 \;, \tag{87}$$

$$\hat{q} = \frac{1}{4}\,\bar{\xi}\;\bar{\eta} \;. \tag{88}$$

(4) Next, the points in the (ξ,η) plane are transformed to the $(\bar{x},\bar{\theta})$ plane by

$$\bar{x}_{ijk} = \ln(\cosh\xi_{ijk} - \cos\eta_{ijk}) \;, \tag{89}$$

$$\theta_{ijk} = \cos^{-1}\left(\frac{1 - \cosh\xi_{ijk}\cos\eta_{ijk}}{\cosh\xi_{ijk} - \cos\eta_{ijk}}\right) \;. \tag{90}$$

(5) The final step is to compute the cylindrical coordinates of each grid point from:

$$x_{ijk} = \bar{x}_{ijk} + d_s - \ln 2 \;, \tag{91}$$

$$\theta_{ijk} = \frac{1}{4}\,\theta_{ijk} \;, \tag{92}$$

$$r_k = R_c + (R_t - R_c)\bar{r}_k \;. \tag{93}$$

3.2 Features of the Grid

The shearing transformation applied at the fourth stage necessarily produces a nonorthogonal grid in the $(\bar{x},\bar{\theta})$ plane. The nonorthogonality is smallest on the lower shearing boundary, under most conditions, and largest at the airfoil surface on the upper shearing boundary, as can be seen from Fig. 5. On the upper (airfoil) boundary the nonorthogonality near the leading edge ($\bar{\xi} = 0$) can be controlled by proper location of the singularity of the unwrapping transformation, Eq. (6). Away from the leading edge the only control over nonorthogonality is to keep the airfoil reasonably thin, say eight percent or less, which will maintain $\bar{\eta}_U$ as close to the image of $\eta = \pi$ as possible.

The parameter which controls grid orthogonality near the airfoil leading edge is d_s in Eq. (3). The most nearly orthogonal system in this region is produced when the leading edge maps into an η = constant line. In the (x,θ) plane such a line is closely approximated by a parabola centered about $\theta = 0$ and is effectively characterized by its radius of curvature at the origin, given by

$$\rho_o = \frac{1}{\left(\frac{d^2x}{d\theta^2}\right)_{\theta=0}} \quad . \tag{94}$$

We determine ρ_o by setting $\eta = \eta_{LE}$ = constant in Eqs. (15) and (16), differentiating the result twice with respect to θ to find $d^2\bar{x}/d\bar{\theta}^2$, plus noting that $d\bar{x}/d\bar{\theta} = 0$ at $\theta = 0$ and by virtue of Eqs. (3) and (5) that

$$\frac{d^2x}{d\theta^2} = 16\,\frac{d^2\bar{x}}{d\bar{\theta}^2} \quad .$$

The result is

$$\rho_o = -\frac{1}{16}\frac{\sin^2 \eta_{LE}}{\cos \eta_{LE}} e^{-\bar{x}_{LE}} \quad . \tag{95}$$

From Eq. (3) evaluated at the airfoil leading edge (x = 0) we have

$$\bar{x}_{LE} = \ln 2 - d_s \tag{96}$$

and from Eq. (17) with $\xi = 0$ and $\eta = \eta_{LE}$ we find that

$$\cos \eta_{LE} = 1 - 2e^{-d_s} \tag{97}$$

from which it follows that

$$\sin \eta_{LE} = 2[e^{-d_s}(1 - e^{-d_s})]^{1/2} \tag{98}$$

Hence, Eq. (95) for ρ_o becomes

$$\rho_o = \frac{1}{8}\left(\frac{1 - e^{-d_s}}{2e^{-d_s} - 1}\right) \quad , \tag{99}$$

which can be solved for d_s to yield,

$$d_s = \ln\left(\frac{1 + 16\,\rho_o}{1 + 8\,\rho_o}\right) \quad . \tag{100}$$

Next, we fit the airfoil leading edge by an osculating parabola, viz.

$$x = K\theta^2 \tag{101}$$

where $K = x_i/\theta_i^2$ and (x_i,θ_i) are appropriate airfoil coordinates near the leading edge. The radius of curvature of the airfoil at the leading edge is, from Eq. (101),

$$\rho_{LE} = \frac{1}{x_o''} = \frac{\theta_i^2}{2x_i} \quad . \tag{102}$$

The optimum value of d_s (which produces the most nearly orthogonal grid near $\bar{\xi} = 0$) is obtained by equating ρ_{LE} and ρ_o. Thus, d_s can then be determined from Eq. (100). Figure 10 shows the variation of $\bar{\eta}_U$ with $\bar{\xi}$ for a six percent thick Joukowsky airfoil for three values of d_s, one of which was determined by Eqs. (100) and (102). In these three cases, we have $d_s << d_L$ which has the effect of limiting the influence of d_s on $\bar{\eta}_U$ to the region $0 < \bar{\xi} < \bar{\xi}_o$ where here $\bar{\xi}_o \cong 0.87$. As r increases from R_c to R_t the leading edge radius of curvature of the airfoil decreases because θ_F decreases--see Eq. (45). Thus d_s must be decreased accordingly.

On the lower shearing boundary the nonorthogonality arises from the mapping of the initial value line (IVL) by Eq. (7). In the (ξ,η) plane the IVL is very nearly half of an ellipse with the ratio of the semi-major to semi-minor axes lengths, defined as $\lambda = \eta_o/\xi_o$ (η_o is the value of η on the IVL at $\xi = 0$) given by

$$\lambda = \frac{\cos^{-1}(1 - 2e^{-a})}{\cosh^{-1}(1 + 2e^{-a})} \quad . \tag{103}$$

Figure 11, in which λ is plotted versus "a", shows that as "a" becomes large λ approaches unity and therefore the IVL approaches a semi-circle in the (ξ,η) plane. Thus for $\bar{\eta}_L$ to have the smallest maximum (at $\bar{\xi} = 0$) and hence for $\bar{\xi}$ = constant lines at $\bar{\eta} = \bar{\eta}_L$ to be as nearly orthogonal as possible, "a" should be large, say 3 or 4, a circumstance desirable on physical grounds anyway.

At the image of the airfoil trailing edge in the $(\bar{\xi},\bar{\eta})$ plane (Points D and E in Fig. 5) when the trailing edge angle is finite the derivative of $\bar{\eta}_U$ with respect to $\bar{\xi}$ will be discontinuous. At the ends of the IVL (Points A and C in Fig. 5) the behavior of $\bar{\eta}_L$ is cusp-like which means that the second derivative of $\bar{\eta}_L$ with respect to $\bar{\xi}$ is discontinuous. These discontinuities produce similar type discontinuities in Y = constant lines via the shearing transformation. This behavior is one of the disadvantages of algebraic mappings involving shearing transformation which is absent in grids generated by solving elliptic partial differential equations. The discontinuous behavior of derivatives of Y = constant lines in the physical plane should therefore be accounted for in the calculation of affected metric coefficients and in the numerical method of solution of the viscous flow equations.

3.3 Numerical Examples

For simplicity a symmetric Joukowsky airfoil was used in the numerical examples of the grid. The ordinates of this airfoil (for unit chord) are given by,

$$y_F = \frac{4\tau}{3\sqrt{3}}(1 - x_F)[4\ x_F(1 - x_F)]^{1/2} \quad , \tag{104}$$

where x_F is measured from the airfoil leading edge and τ is the maximum thickness to chord ratio. Two example grids in the $(\bar{x},\bar{\theta})$ plane are presented with parameters listed in Table 1 below. The parameter J is the number of points in the Y direction.

Parameter	Case 1	Case 2
N_1	---	15
N_2	---	15
J	31	31
d_{IVL}	3.0	3.0
d_s	0.05	0.05
d_{OB}	3.0	3.0
τ	0.12	0.06
R_c	1.0	1.0
r/R_c	1.0	1.0
S_0	10	10
S_1	10	10
$\hat{S}_0$	---	0.2
$\hat{C}$	---	1.2

Table 1. Grid parameters for Numerical Examples

Case 1 is shown in Fig. 12 and Case 2 in Fig. 13. Case 1 has no stretching function in X and no X = constant line through the corner. The non-orthogonality of the grid in Case 1 (12% thick) is seen to be more pronounced at the airfoil surface than in Case 2 (6% thick) which bears out the remark made earlier. Notice that both examples are for the grid in the $(\bar{x},\bar{\theta})$ plane on the cylinder surface ($r = R_c$) which corresponds to the intersection of the fin with the cylinder. Hence in these examples, by Eq. (45), the airfoil thickness in terms of $\bar{\theta}$ is a maximum and thus the nonorthogonality is most pronounced.

The computer code listing is given in the appendix.

References

(1) Jameson, A., "Iterative Solution of Transonic Flows over Airfoils and Wings, Including Flow at Mach 1," Comm. Pure Appl. Math. 27, 283-309 (1974).

(2) Caughey, D. A. and A. Jameson, "Numerical Calculation of Transonic Potential Flow about Wing-Body Combinations," AIAA Jour. 17, 175-181 (1979).

(3) Caughey, D. A. and A. Jameson, "Progress in Finite-Volume Calculations for Wing-Fuselage Combinations," AIAA Jour. 18, 1281-1288 (1980).

(4) Caughey, D. A., "A Systematic Procedure for Generating Useful Conformal Mappings," Int. Jour. Numer. Meth. Engin. 12, 1651-1657 (1978).

(5) Vinokur, M., "On One-Dimensional Stretching Functions for Finite-Difference Calculations," NASA CR-3313 (October 1980).

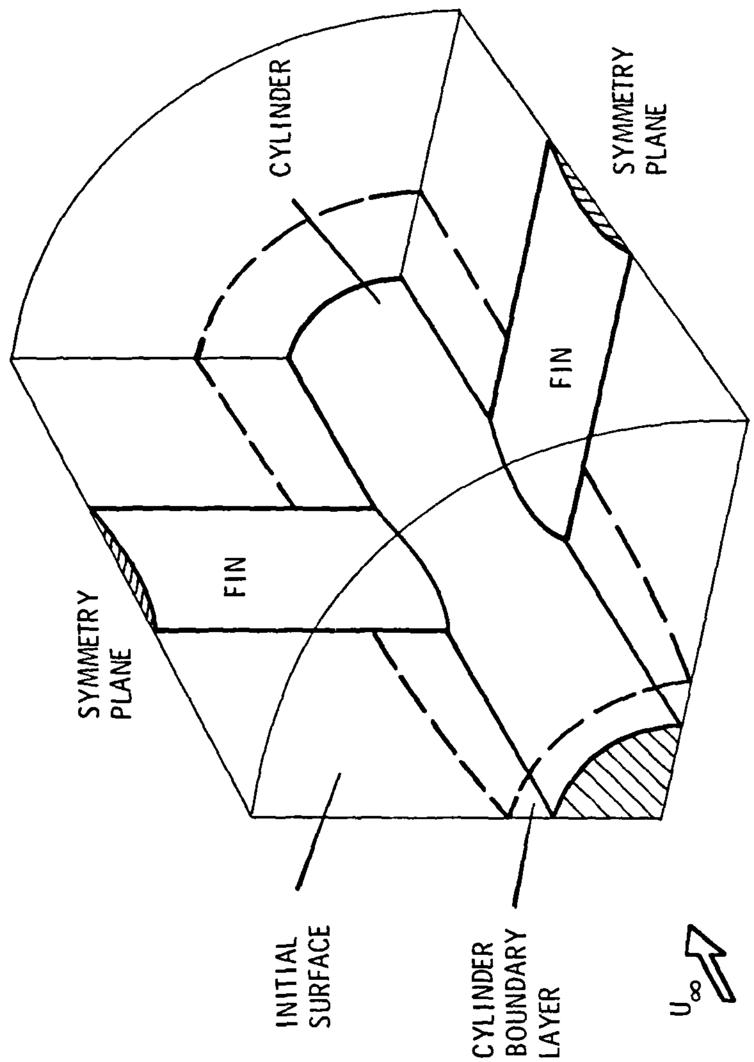


Figure 1. Schematic of Geometry and Computational Domain.

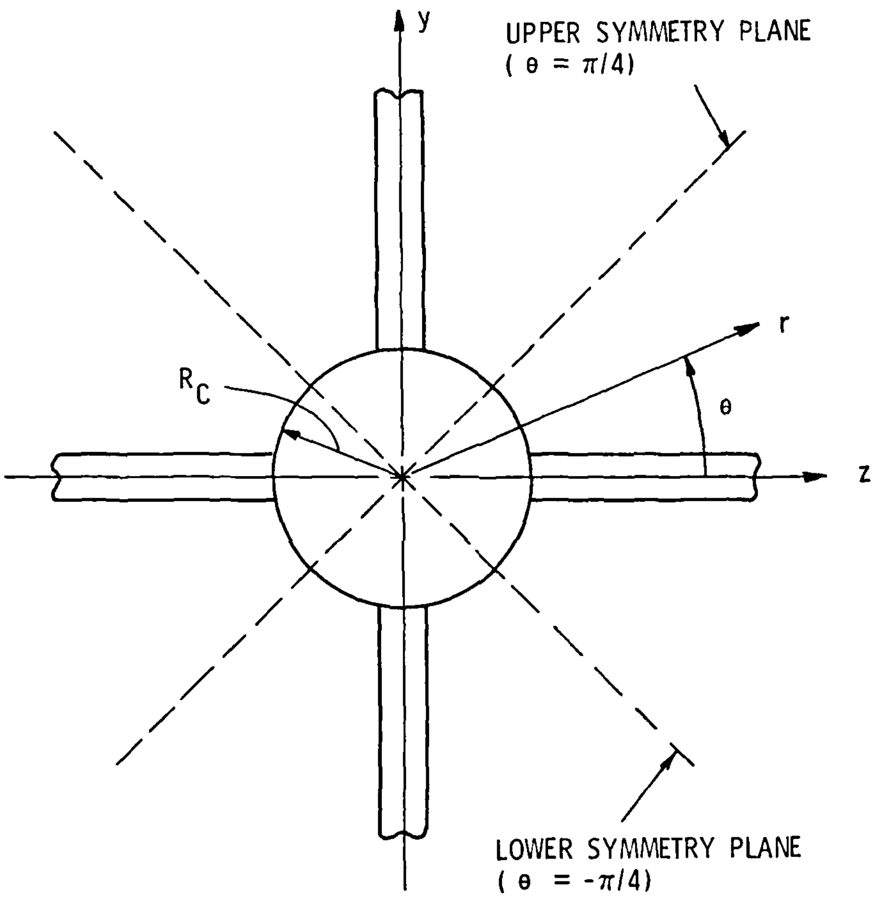


Figure 2. Coordinate System in Crossflow Plane.

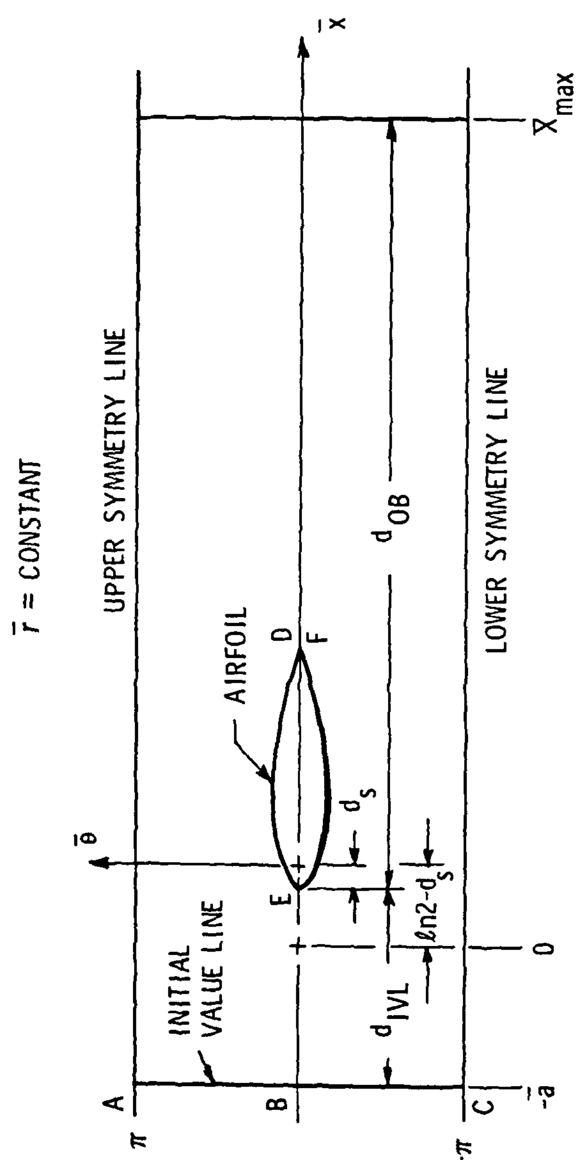


Figure 3. Computational Domain in $(\bar{x},\bar{\theta})$ Plane.

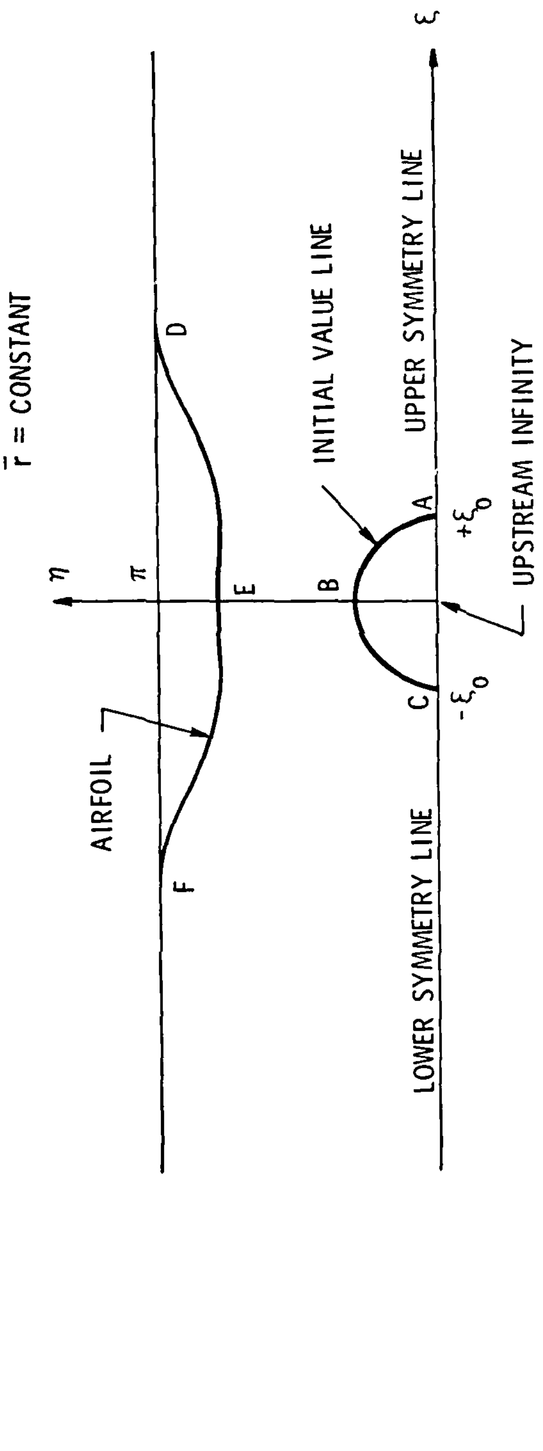


Figure 4. Boundary Images in (ξ,η) Plane.

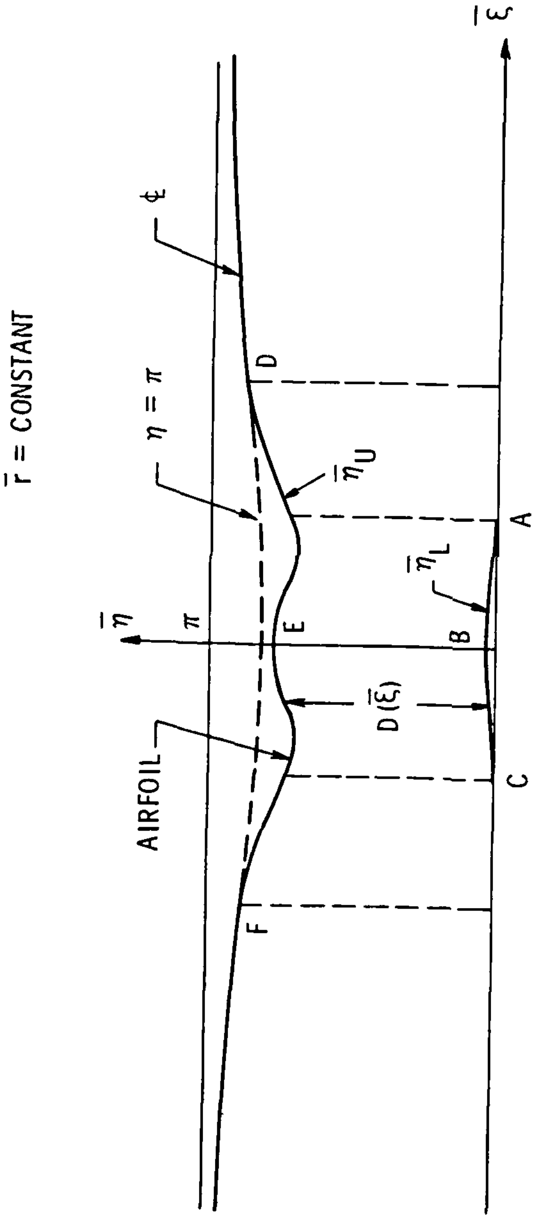


Figure 5. Boundary Images in $(\bar{\xi},\bar{\eta})$ Plane.

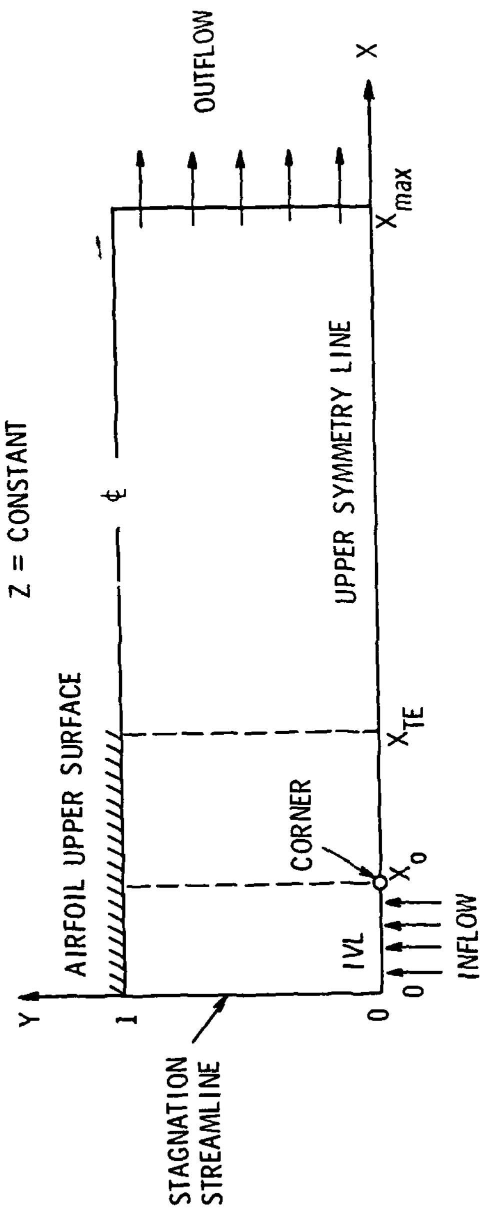


Figure 6. Schematic of Computational Plane.

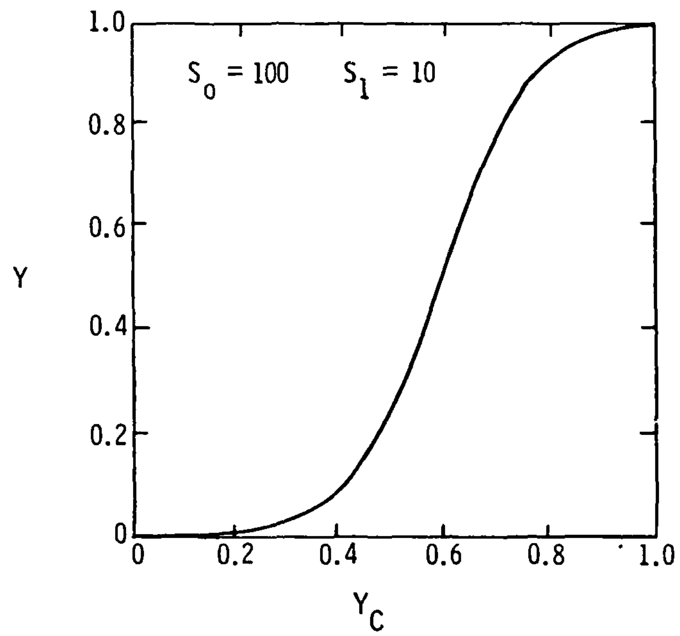


Figure 7. Two-Sided Stretching Function for Y.

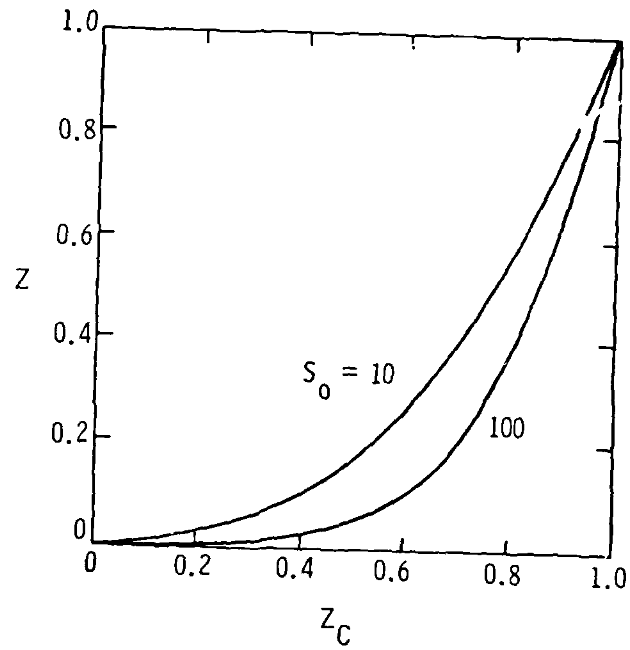


Figure 8. One-Sided Stretching Function for Z.

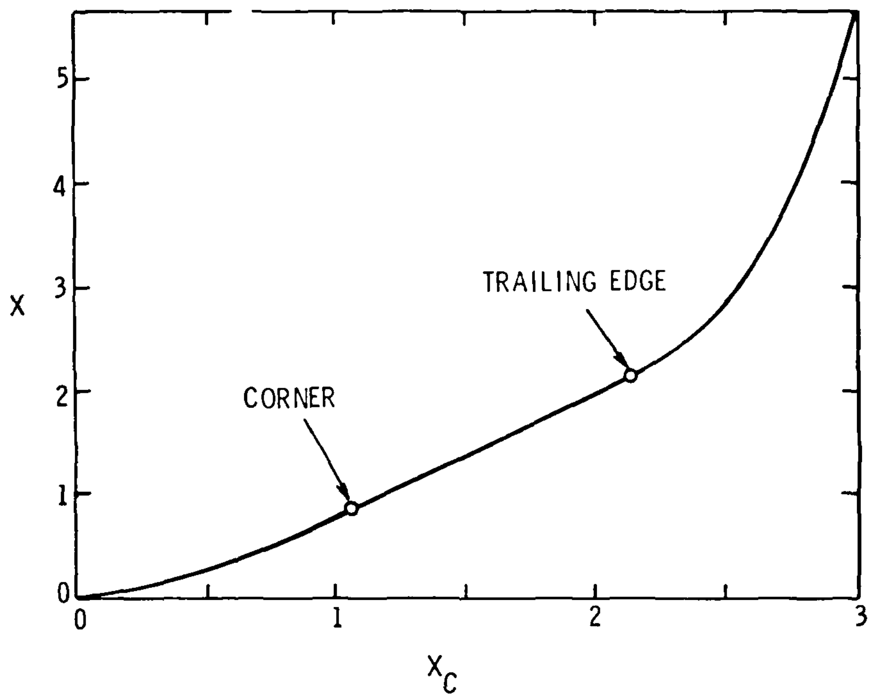


Figure 9. Three Segment Stretching Function for X.

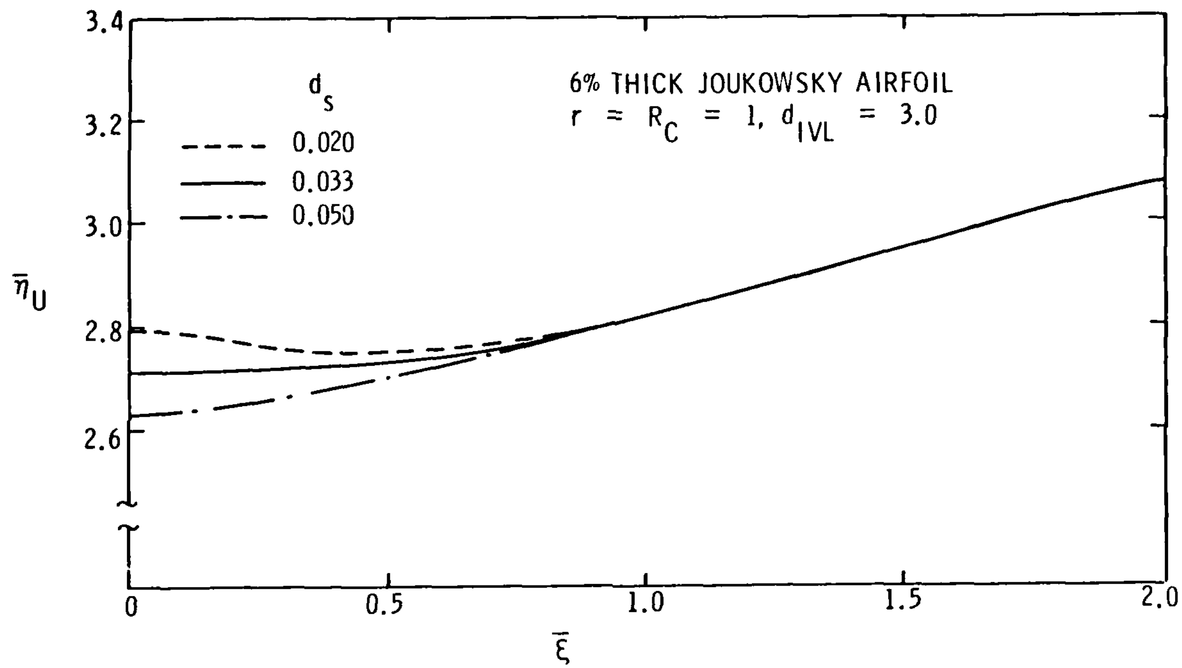


Figure 10. Effect of Singularity Location on Upper Shearing Boundary.

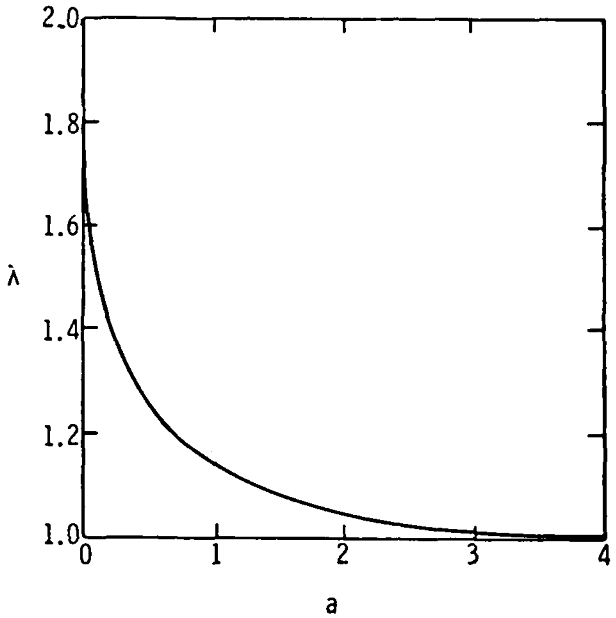


Figure 11. Ellipticity of Initial Value Line in (ξ,η) Plane.

30 March 1983
GHH:lhm

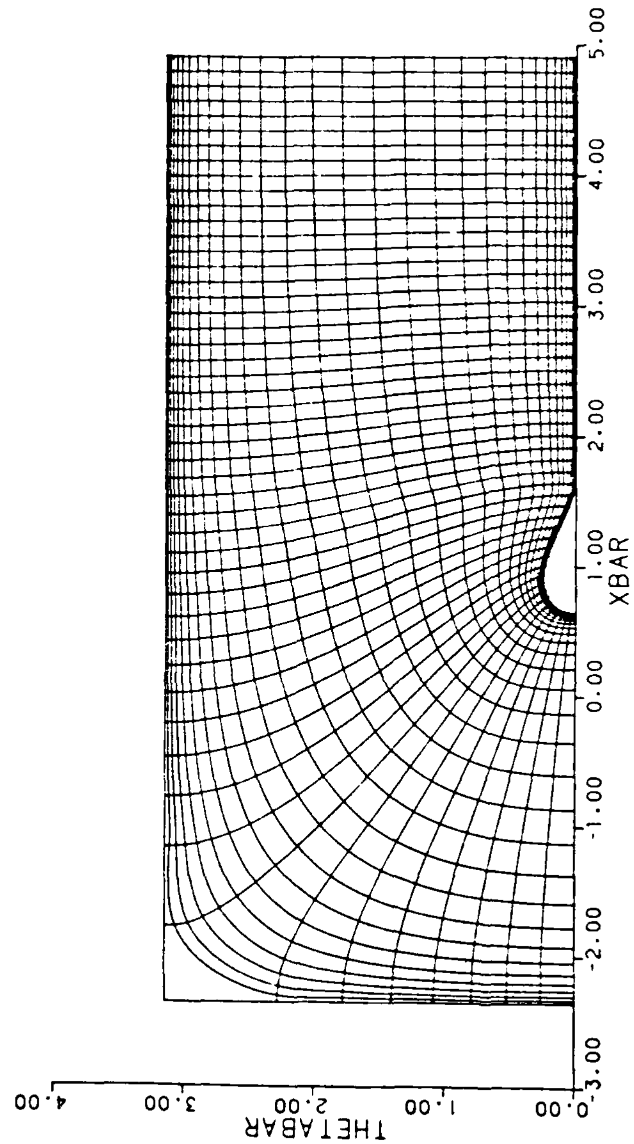


Figure 12. Grid for a Twelve Percent Thick Joukowsky Airfoil.

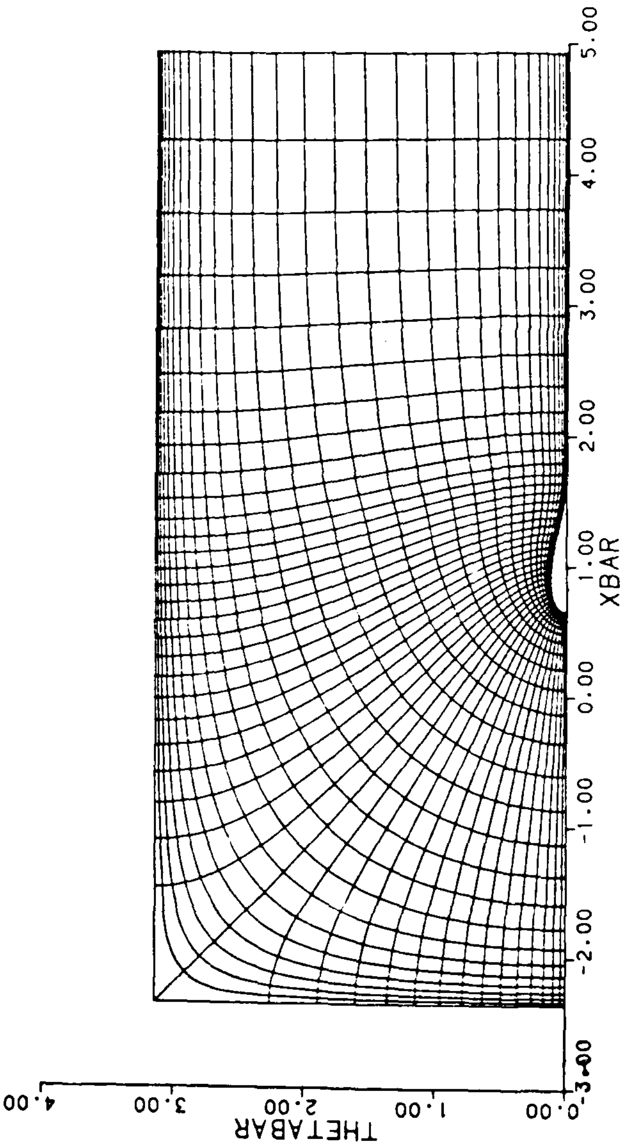


Figure 13. Grid for a Six Percent Thick Joukowsky Airfoil.

Appendix: Grid Generation Computer Code Listing

```
C      PROGRAM NAME: CGRID3
C      THIS PROGRAM COMPUTES A SURFACE FITTED C-GRID FOR A FIN
C      CYLINDER BODY.
C      THE FIN IN THIS VERSION IS A SYMMETRIC JOUKOWSKY AIRFOIL.
C      THIS IS THE 3-D VERSION.
C*****************************************************************
       IMPLICIT REAL*8 (A-H,O-Z)
       COMMON /BLK01/ IMAX,JMAX,ITE,ITEM,ILAST,ISEG1,ISEG2
       COMMON /BLK02/ XIBM,XIO,XIBO
       COMMON /BLK03/ C1,C2,C3,C4,C5,PI,PISQ
       COMMON /BLK04/ XF(101),YF(101)
       COMMON /BLK05/ XIB(151),SBAR(151),ETABL(151)
       COMMON /BLK06/ SY0,SY1,SZ0,SX0,SSR
       COMMON /BLK07/ ZC(151),BIGZ(151)
C
     1 FORMAT(5I4)
     2 FORMAT(5F10.4)
    10 FORMAT(1H1,4X,'INPUT PARAMETERS FOR C-GRID')
    11 FORMAT(1H0,9X,'ISEG1 =',I6/10X,'ISEG2 =',I6/10X,'JMAX  =',
      1I6/10X,'KMAX  ='I6/10X,'ITE   =',I6)
    12 FORMAT(10X,'DIVL =',F10.4/10X,'DOB =',F10.4/
      110X,'TAU  =',F10.4/10X,'RC   =',F10.4/10X,'RT   =',F10.4/10X,
      2'SY0  =',F10.4/10X,'SY1  =',F10.4/10X,'SZ0  =',F10.4/10X,'SX0  =',
      3F10.4/10X,'SSR  =',F10.4)
    13 FORMAT(1H0)
    14 FORMAT(5X,'STACKED C-GRID FOR FIN-CYLINDER GEOMETRY')
    15 FORMAT(10X,'DS   =',D14.4)
C
C      INPUT REQUIREMENTS
C
C           ISEG1 = NO. INTERVALS ON FIRST X-SEGMENT.
C           ISEG2 = NO. INTERVALS ON SECOND X-SEGMENT.
C           IMAX  = NO. POINTS IN X-DIRECTION.
C           JMAX  = NO. POINTS IN Y-DIRECTION.
C           KMAX  = NO. POINTS IN Z-DIRECTION.
C           ITE   = NO. POINTS ON AIRFOIL INITIALLY.
C           DIVL  = DISTANCE FROM AIRFOIL L.E. TO INITIAL VALUE LINE.
C           DS    = DISTANCE FROM AIRFOIL L.E. TO SINGULARITY OF
C                   COORDINATE SYSTEM.
C           DOB   = DISTANCE FROM AIRFOIL T.E. TO OUTFLOW BOUNDARY.
C           TAU   = AIRFOIL MAX. THICKNESS TO CHORD RATIO.
C           RC    = INNER CYLINDER RADIUS, IN TERMS OF AIRFOIL CHORD.
C           RT    = OUTER CYLINDER RADIUS, IN TERMS OF AIRFOIL CHORD.
C           SY0   = Y-STRETCHING PARAMETER AT AIRFOIL SURFACE.
C           SY1   = Y-STRETCHING PARAMETER AT INITIAL SURFACE.
C           SZ0   = Z-STRETCHING PARAMETER AT INNER CYLINDER.
C           SX0   = INITIAL X-STRETCHING PARAMETER, SEGMENT 1.
C           SSR   = X-GEOMETRIC PROGRESSION RATIO, SEGMENT 3.
C
       READ(5,1) ISEG1,ISEG2,JMAX,KMAX,ITE
       READ(5,2) DIVL,DOB
       READ(5,2) TAU,RC,RT
       READ(5,2) SY0,SY1,SZ0,SX0,SSR
       ITEM=ITE-1
       WRITE(6,10)
       WRITE(6,11) ISEG1,ISEG2,JMAX,KMAX,ITE
       WRITE(6,12) DIVL,DOB,TAU,RC,RT,SY0,SY1,SZ0,SX0,SSR
       WRITE(6,13)
       WRITE(6,14)
C
       C3=2.0D0*TAU/DSQRT(27.0D0)
```

```
      PI=3.14159265358979D0
      PISQ=PI*PI
      XE=1.0D0+D08
C
C     CALCULATE AIRFOIL COORDINATES.
C
      CALL FOIL
C
C     CALCULATE ZC AND BIGZ.
C
      CALL STRFZ(ZC,BIGZ,KMAX,SZ0)
      DELR=RT-RC
C
C     BEGIN CALCULATION OF STACKED GRID.
C
      DO 50 K=1,KMAX
      RAD=RC+DELR*BIGZ(K)
C
C     CALCULATE DS - DISTANCE FROM AIRFOIL LEADING EDGE TO
C     SINGULARITY OF UNWRAPPING TRANSFORMATION.
C
      THF=DASIN(YF(4)/RAD)
      RHO=0.5D0*THF*THF/XF(4)
      DS=DLOG((1.0D0+16.0D0*RHO)/(1.0D0+8.0D0*RHO))
      WRITE(6,13)
      WRITE(6,15) DS
      WRITE(6,13)
      C1=DEXP(-(DIVL+DS))
      C2=2.0D0*C1
      RHS=DSQRT(4.0D0*C1*(1.0D0+C1))
      CALL ASINH(XI0,RHS)
      C4=DLOG(2.0D0)-DS
      C5=XI0*XI0
      XIB0=2.0D0*XI0
C
C     CALCULATE XIBM - COORDINATE OF DOWNSTREAM BOUNDARY IN XI BAR -
C     ETA BAR PLANE.
C
      XBE=XE+C4
      TERM=DEXP(XBE)-1.0D0
      RHS=DSQRT(TERM*TERM-1.0D0)
      CALL ASINH(XIE,RHS)
      XIBM=XIE*(1.0D0+C5/(PISQ+XIE*XIE))
C
      CALL SHEAR(RAD)
      KK=K
      CALL XTGRID(KK,RAD)
   50 CONTINUE
      STOP
      END
      SUBROUTINE SHEAR(RAD)
C**********************************************************************
C     THIS SUBROUTINE CALCULATES SBAR VS. XI BAR, TO BE USED IN THE
C     SHEARING TRANSFORMATION.
C**********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /BLK01/ IMAX,JMAX,ITE,ITEM,ILAST,ISEG1,ISEG2
      COMMON /BLK02/ XIBM,XI0,XIB0
      COMMON /BLK03/ C1,C2,C3,C4,C5,PI,PISQ
      COMMON /BLK04/ XF(101),YF(101)
      COMMON /BLK05/ XIB(151),SBAR(151),ETABL(151)
```

```
C
      DIMENSION ETABU(151)
C
   10 FORMAT(5X,'SHEARING BOUNDARY IN XIBAR - ETABAR PLANE')
   11 FORMAT(1H0)
   12 FORMAT(5X,'I',6X,'XIBAR',9X,'ETABL',9X,'ETABU',9X,'SBAR')
   13 FORMAT(I6,4D14.4)
   14 FORMAT(1H0,4X,'UNABLE TO CONVERGE XI IN 50 ITERATIONS'/5X,
     1'XIBAR =',D14.4)
C
C     COMPUTE NORMALIZED AIRFOIL COORDINATES FOR GIVEN CYLINDRICAL
C     RADIUS AND TRANSFORM TO XI BAR - ETA BAR PLANE.  THIS STEP
C     GIVES THE FIRST PORTION OF THE UPPER BOUNDARY.
C
      WRITE(6,11)
      WRITE(6,10)
      WRITE(6,11)
      T1=RAD*RAD
      DO 50 I=1,ITE
      YFI=YF(I)
      THF=DASIN(YFI/RAD)
      XBFI=XF(I)+C4
      THBFI=4.0D0*THF
C
      T2=DEXP(XBFI)
      PBAR=1.0D0-T2*DCOS(THBFI)
      QBAR=T2*DSIN(THBFI)
      QSQ=QBAR*QBAR
      BETA=1.0D0-PBAR*PBAR-QSQ
      ALPHA=DSQRT(BETA*BETA+4.0D0*QSQ)
      RHS=DSQRT(0.5D0*(ALPHA-BETA))
      CALL ASINH(XIF,RHS)
      ARG=PBAR/DCOSH(XIF)
      IF((ARG+1.0D0).LT.0.0D0) ARG=-1.0D0
      ETAF=DACOS(ARG)
C
      XMU=C5/(XIF*XIF+ETAF*ETAF)
      XIB(I)=XIF*(1.0D0+XMU)
      ETABU(I)=ETAF*(1.0D0-XMU)
   50 CONTINUE
C
C     CONTINUE UPPER BOUNDARY CALCULATION BEYOND AIRFOIL T.E.
C     TO XIBM.
C
      DXIB=0.2D0
      ILAST=ITE+(XIBM-XIB(ITE))/DXIB
      IF(ILAST.GT.151) ILAST=151
      ITEP=ITE+1
      WRITE(6,12)
      DO 100 I=ITEP,ILAST
      XIBAR=XIB(I-1)+DXIB
      XIB(I)=XIBAR
      XIL=XIBAR
      DO 70 IT=1,50
      XMU=C5/(PISQ+XIL*XIL)
      XI=XIBAR/(1.0D0+XMU)
      IF(DABS(XI-XIL).LT.1.0D-08) GO TO 80
   70 XIL=XI
      WRITE(6,14) XIBAR
      STOP
   80 ETABU(I)=PI*(1.0D0-XMU)
```

```
  100 CONTINUE
C
C     CALCULATE LOWER BOUNDARY IN XI BAR - ETA BAR PLANE AND
C     SBAR.
C
      DO 200 I=1,ILAST
      XIBAR=XIB(I)
      IF(XIBAR.GE.XIB0) GO TO 140
      XIL=XIBAR
      XMU=1.0D0
      DO 120 IT=1,50
      XI=XIBAR/(1.0D0+XMU)
      ARG=DCOSH(XI)-C2
      ETA=DACOS(ARG)
      XMU=C5/(XI*XI+ETA*ETA)
      IF(DABS(XI-XIL).LT.1.0D-08) GO TO 130
  120 XIL=XI
      WRITE(6,14) XIBAR
      STOP
  130 ETABL(I)=ETA*(1.0D0-XMU)
      GO TO 150
  140 ETABL(I)=0.0D0
  150 SBAR(I)=ETABU(I)-ETABL(I)
      WRITE(6,13) I,XIB(I),ETABL(I),ETABU(I),SBAR(I)
  200 CONTINUE
      RETURN
      END
      SUBROUTINE XIGRID(K,RAD)
C*****************************************************************
C     THIS SUBROUTINE CALCULATES THE GRID IN THE XBAR - THETABAR
C     PLANE.
C*******************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /BLK01/ IMAX,JMAX,ITE,ITEM,ILAST,ISEG1,ISEG2
      COMMON /BLK02/ XIBM,XI0,XIB0
      COMMON /BLK03/ C1,C2,C3,C4,C5,PI,PISQ
      COMMON /BLK05/ XIB(151),SBAR(151),ETABL(151)
      COMMON /BLK06/ SY0,SY1,SZ0,SX0,SSR
      COMMON /BLK07/ ZC(151),BIGZ(151)
C
      DIMENSION BIGX(151),BIGY(151)
      DIMENSION XC(151),YC(151)
C
   11 FORMAT(1H0)
   12 FORMAT(5X,'I',5X,'J',5X,'K',6X,'XC',12X,'YC',12X,'ZC',12X,
     1'R',13X,'X',13X,'THETA')
   13 FORMAT(3I6,6D14.4)
C
C     SET UP GRID IN COMPUTATIONAL PLANE.
C
      XBTE=XIB(ITE)
      NPTE=ISEG1+ISEG2+1
      CALL STRFX(XC,BIGX,ISEG1,ISEG2,IMAX,SX0,SSR,XIB0,XBTE,XIBM)
      CALL STRFY(YC,BIGY,JMAX,SY0,SY1)
C
C     DETERMINE GRID IN PHYSICAL PLANE.
C
      I=1
      XIBAR=0.0D0
      SBI=SBAR(1)
      ETBLI=ETABL(1)
```

```
      WRITE(6,11)
      WRITE(6,12)
      WRITE(6,11)
      DO 70 J=1,JMAX
      ETABAR=ETBLI+SBI*BIGY(J)
      P=0.25D0*ETABAR*ETABAR+C5
      XI=0.0D0
      ETA=0.5D0*ETABAR+DSQRT(P)
      XBAR=DLOG(1.0D0-DCOS(ETA))
      THBAR=0.0D0
      XX=XBAR-C4
      THETA=0.0D0
      WRITE(6,13) I,J,K,XC(I),YC(J),ZC(K),RAD,XX,THETA
   70 CONTINUE
      IBEG=1
      IEND=ITE
      DO 100 I=2,IMAX
      XIBAR=BIGX(I)
      IF(I.LE.NPTE) GO TO 80
      IBEG=ITE
      IEND=ILAST
C
C     INTERPOLATE TO FIND SBAR AND ETABL CORRESPONDING TO XIBAR.
C
   80 CALL INTERP(XIB,SBAR,XIBAR,SBI,IBEG,IEND,INT,0)
      CALL INTERP(XIB,ETABL,XIBAR,ETBLI,IBEG,IEND,INT,1)
C
      WRITE(6,11)
      WRITE(6,12)
      WRITE(6,11)
      DO 100 J=1,JMAX
      ETABAR=ETBLI+SBI*BIGY(J)
      Q=0.25D0*XIBAR*ETABAR
      P=0.25D0*(XIBAR*XIBAR-ETABAR*ETABAR)-C5
      XMU=DSQRT(P*P+4.0D0*Q*Q)
      XI=0.5D0*XIBAR+DSQRT(0.5D0*(XMU+P))
      ETA=0.5D0*ETABAR+DSQRT(0.5D0*(XMU-P))
C
      T1=DCOSH(XI)
      T2=DCOS(ETA)
      ARG1=T1-T2
      XBAR=DLOG(ARG1)
      THBAR=DACOS((1.0D0-T1*T2)/ARG1)
      THETA=0.25D0*THBAR
      XX=XBAR-C4
      WRITE(6,13) I,J,K,XC(I),YC(J),ZC(K),RAD,XX,THETA
  100 CONTINUE
      RETURN
      END
      SUBROUTINE ASINH(ARG,RHS)
C*****************************************************************
C     THIS SUBROUTINE COMPUTES THE INVERSE HYPERBOLIC SINE USING
C     NEWTON'S METHOD.
C*****************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
C
   10 FORMAT(1H0,4X,'INVERSE HYPERBOLIC SINE CALCULATION FAILED FOR
     1SINH(X) =',D14.7)
C
      TEST=DABS(RHS)
      IF(TEST.GT.1.0D0) GO TO 30
```

```
      ARG=RHS
      GO TO 40
   30 ARG=DLOG(2.0D0*TEST)*DSIGN(1.0D0,RHS)
   40 CONTINUE
      DO 50 K=1,50
      FA=DSINH(ARG)-RHS
      FPA=DCOSH(ARG)
      DARG=-FA/FPA
      IF(DABS(DARG).LT.1.0D-10) RETURN
      ARG=ARG+DARG
   50 CONTINUE
      WRITE(6,10) RHS
      RETURN
      END
      SUBROUTINE FOIL
C**********************************************************************
C     THIS SUBROUTINE GENERATES (X,Y) COORDINATES FOR A SYMMETRIC
C     JOUKOWSKY AIRFOIL.
C**********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /BLK01/ IMAX,JMAX,ITE,ITEM,ILAST,ISEG1,ISEG2
      COMMON /BLK03/ C1,C2,C3,C4,C5,PI,PISQ
      COMMON /BLK04/ XF(101),YF(101)
C
   10 FORMAT(5X,'AIRFOIL COORDINATES')
   11 FORMAT(1H0)
   12 FORMAT(5X,'I',6X,'XF',13X,'YF')
   13 FORMAT(I6,2D14.4)
C
      DTH=PI/ITEM
      XF(1)=0.0D0
      YF(1)=0.0D0
      DO 50 I=2,ITEM
      TH=(I-1)*DTH
      T1=DCOS(TH)
      XF(I)=0.5D0*(1.0D0-T1)
      YF(I)=C3*(1.0D0+T1)*DSIN(TH)
   50 CONTINUE
      XF(ITE)=1.0D0
      YF(ITE)=0.0D0
      WRITE(6,11)
      WRITE(6,10)
      WRITE(6,11)
      WRITE(6,12)
      DO 60 I=1,ITE
   60 WRITE(6,13) I,XF(I),YF(I)
      RETURN
      END
      SUBROUTINE INTERP(XX,YY,XINT,YINT,IBEG,IEND,INT,ISW)
C*******************************************************************
C     THIS SUBROUTINE USES LAGRANGE CUBIC INTERPOLATION TO
C     DETERMINE YINT FOR A GIVEN XINT.
C
C           XX   = INDEPENDENT VARIABLE.
C           YY   = DEPENDENT VARIABLE.
C           IBEG = INITIAL INDEX FOR INTERPOLATION RANGE.
C           IEND = FINAL INDEX FOR INTERPOLATION RANGE.
C           INT  = UPPER INDEX OF INTERPOLATION INTERVAL.
C           ISW  = INTERPOLATION INTERVAL SEARCH SWITCH.
C                  0  PERFORM SEARCH.
C                  1  OMIT SEARCH.
```

```
C*******************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
C
      DIMENSION XX(151),YY(151)
C
      IF(ISW.GT.0) GO TO 75
   60 DO 70 I=IBEG,IEND
      INT=I
      IF(XX(I).GT.XINT) GO TO 75
   70 CONTINUE
   75 IF(INT.EQ.(IBEG+1)) GO TO 80
      IF(INT.EQ.IEND) GO TO 90
      I1=INT-2
      I2=INT-1
      I3=INT
      I4=INT+1
      GO TO 100
   90 I1=IBEG
      I2=IBEG+1
      I3=IBEG+2
      I4=IBEG+3
      GO TO 100
   90 I1=IEND-3
      I2=IEND-2
      I3=IEND-1
      I4=IEND
  100 CONTINUE
      X1=XX(I1)
      X2=XX(I2)
      X3=XX(I3)
      X4=XX(I4)
      CF1=(XINT-X2)*(XINT-X3)*(XINT-X4)/((X1-X2)*(X1-X3)*(X1-X4))
      CF2=(XINT-X1)*(XINT-X3)*(XINT-X4)/((X2-X1)*(X2-X3)*(X2-X4))
      CF3=(XINT-X1)*(XINT-X2)*(XINT-X4)/((X3-X1)*(X3-X2)*(X3-X4))
      CF4=(XINT-X1)*(XINT-X2)*(XINT-X3)/((X4-X1)*(X4-X2)*(X4-X3))
      YINT=CF1*YY(I1)+CF2*YY(I2)+CF3*YY(I3)+CF4*YY(I4)
      RETURN
      END
      SUBROUTINE STRFY(XI,T,NPT,SY0,SY1)
C*******************************************************************
C     THIS SUBROUTINE GENERATES A NONUNIFORM POINT DISTRIBUTION
C     USING VINOKURS TWO-SIDED STRETCHING FUNCTION, AS GIVEN IN
C     NASA CR-3133.
C*******************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
C
      DIMENSION XI(151),T(151)
C
C     COMPUTE XI.
C
      DXI=1.0D0/(NPT-1)
      DO 40 J=1,NPT
   40 XI(J)=(J-1)*DXI
C
C     COMPUTE DELTA Y.
C
      A=DSQRT(SY0/SY1)
      B=DSQRT(SY0*SY1)
      TEST=2.7829681D0
      IF(B.GT.TEST) GO TO 50
      YBAR=B-1.0D0
```

```
      DELY=(((((-0.0010794123D0*YBAR+0.0077424461D0)*YBAR
     1-0.024907295D0)*YBAR+0.057321429D0)*YBAR-0.15D0)*YBAR
     2+1.0D0)*DSQRT(6.0D0*YBAR)
      GO TO 60
   50 V=DLOG(B)
      W=1.0D0/B-0.028527431D0
      DELY=(((8.56795911D0*W-2.6294547D0)*W+1.9496443D0)*W
     1+0.24902722D0)*W-0.02041793D0+V+(1.0D0+1.0D0/V)*
     2DLOG(2.0D0*V)
   60 CONTINUE
C
C     COMPUTE T.
C
      C1=A*DSINH(DELY)
      C2=1.0D0-A*DCOSH(DELY)
      DO 70 I=1,NPT
      FN=DTANH(DELY*XI(I))
      T(I)=FN/(C1+C2*FN)
   70 CONTINUE
      RETURN
      END
      SUBROUTINE STRFX(XI,T,NSEG1,NSEG2,NMAX,SXO,SSR,XIBO,XBTE,XIBM)
C*********************************************************************
C     THIS SUBROUTINE GENERATES A NONUNIFORM POINT DISTRIBUTION
C     SPECIALIZED TO THE COORDINATE WRAPPED AROUND THE AIRFOIL.
C*********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
C
      DIMENSION XI(151),T(151)
C
C     SEGMENT NUMBER 1.
C
      TTE=XBTE/XIBO
      TMAX=XIBM/XIBO
      DXI=1.0D0/NSEG1
      NP1=NSEG1+1
      S1=0.5D0*(SXO-1.0D0)
      DO 50 I=1,NP1
      XX=(I-1)*DXI
      XI(I)=XX
   50 T(I)=XX*(1.0D0+S1*(1.0D0-XX)*(2.0D0-XX))
C
C     SEGMENT NUMBER 2.
C
      AA=0.5D0*(3.0D0-SXO)
      XWTE=NSEG2*DXI
      BB=(TTE-1.0D0-AA*XWTE)/(XWTE*XWTE)
      NP2=NSEG2+1
      DO 60 K=2,NP2
      I=NSEG1+K
      XW=(K-1)*DXI
      XI(I)=1.0D0+XW
   60 T(I)=1.0D0+XW*(AA+XW*BB)
C
C     SEGMENT NUMBER 3.
C
      N3=NSEG1+NSEG2
      NP3=N3+1
      XITE=XI(NP3)
      DT1=T(NP3)-T(N3)
      S1=DT1/(SSR-1.0D0)
```

```
      KMAX=151-NP3
      DO 70 K=2,KMAX
      I=N3+K
      XI(I)=XITE+(K-1)*DXI
      TI=TTE+S1*(SSP**(K-1)-1.0D0)
      T(I)=TI
      IF(TI.GE.TMAX) GO TO 80
   70 CONTINUE
   80 NMAX=I
C
C     RESCALE VARIABLES.
C
      SCALE=XBTE/XITE
      DO 90 I=1,NMAX
      XI(I)=SCALE*XI(I)
   90 T(I)=XIB0*T(I)
      RETURN
      END
      SUBROUTINE STRFZ(XI,T,NPT,S0)
C*****************************************************************
C     THIS SUBROUTINE GENERATES A NONUNIFORM POINT DISTRIBUTION USING
C     VINOKURS ONE-SIDED STRETCHING FUNCTION.
C*****************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
C
      DIMENSION XI(151),T(151)
C
C     COMPUTE XI.
C
      DXI=1.0D0/(NPT-1)
      DO 40 K=1,NPT
   40 XI(K)=(K-1)*DXI
C
C     COMPUTE DELTA Y.
C
      TEST=2.7829681D0
      IF(S0.GT.TEST) GO TO 50
      YBAR=S0-1.0D0
      DELY=(((((-0.0010794123D0*YBAR+0.0077424461D0)*YBAR
     1-0.024907295D0)*YBAR+0.057321429D0)*YBAR-0.15D0)*YBAR
     2+1.0D0)*DSQRT(6.0D0*YBAR)
      GO TO 60
   50 V=DLOG(S0)
      W=1.0D0/S0-0.028527431D0
      DELY=(((8.56795911D0*W-2.6294547D0)*W+1.9496443D0)*W
     1+0.24902722D0)*W-0.02041793D0+V+(1.0D0+1.0D0/V)*
     2DLOG(2.0D0*V)
   60 CONTINUE
C
C     COMPUTE T.
C
      C1=0.5D0*DELY
      C2=1.0D0/DTANH(C1)
      DO 70 K=1,NPT
      T(K)=1.0D0+C2*DTANH(C1*(XI(K)-1.0D0))
   70 CONTINUE
      RETURN
      END
```

DISTRIBUTION LIST FOR UNCLASSIFIED TM 83-45
by G. H. Hoffman, dated 30 March 1983

Office of Naval Research
Department of the Navy
800 North Quincy Street
Arlington, VA 22217
Attn: R. E. Whitehead
(Copy No. 1)

Office of Naval Research
Department of the Navy
800 North Quincy Street
Arlington, VA 22217
Attn: C. Lee
(Copy No. 2)

Commander
Naval Sea Systems Command
Department of the Navy
Washington, DC 20362
Attn: T. E. Peirce
Code NSEA-63R31
(Copy No. 3)

Commander
Naval Underwater Systems Center
Department of the Navy
Newport, RI 02840
Attn: D. J. Goodrich
Code 3634
(Copy No. 4)

Commander
David W. Taylor Naval Ship R&D Center
Department of the Navy
Bethesda, MD 20084
Attn: Library
(Copy No. 5)

Commander
Naval Surface Weapons Center
Department of the Navy
Silver Spring, MD 20910
Attn: Library
(Copy No. 6)

Defense Technical Information Center
5010 Duke Street
Cameron Station
Alexandria, VA 22314
(Copies 7 through 12)

Naval Research Laboratory
Department of the Navy
Washington, DC 20390
Attn: Library
(Copy No. 13)

Superintendent
Code 1424
Naval Post Graduate School
Monterey, CA 93949
(Copy No. 14)

NASA Lewis Research Center
21000 Brookpark Road
Cleveland, OH 44135
Attn: P. M. Sockol
Code MS 77-5
(Copy No. 15)

Professor C. L. Merkle
Department of Mechanical Engineering
The Pennsylvania State University
University Park, PA 16802
(Copy No. 16)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: R. E. Henderson
(Copy No. 17)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: B. R. Parkin
(Copy No. 18)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: S. A. Abdallah
(Copy No. 19)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: M. W. McBride
(Copy No. 20)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: G. C. Lauchle
(Copy No. 21)

Director
Applied Research Laboratory
The Pennsylvania State University
Post OFfice Box 30
State College, PA 16801
Attn: W. S. Gearhart
(Copy No. 22)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: K. C. Kaufman
(Copy No. 23)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: G. H. Hoffman
(Copy No. 24)

Director
Applied Research Laboratory
The Pennsylvania state University
Post Office Box 30
State College, PA 16801
Attn: J. J. Eisenhuth
(Copy No. 25)

Director
Applied Research Laboratory
The Pennsylvania State University
Post Office Box 30
State College, PA 16801
Attn: Garfield Thomas Water Tunnel Files
(Copy No. 26)

TE

MED

8.